

**Amendment to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

Claim 1 (currently amended): A method for providing a requestor with access to dynamic data via quasi-static data requests, comprising the steps of:

- defining a web page, said web page including at least one dynamic element;
- creating an executable digital code to be run on a computer and invoked at defined intervals by a scheduler component, said executable code ~~creating effective to create~~ and storing a quasi-static copy of said defined web page;
- creating said scheduler component capable of invoking said executable code at predefined intervals;
- loading said executable code and said scheduler component onto a platform in connectivity with a web server and with one another;
- invoking execution of said scheduler component; and
- retrieving and returning the static copy of said defined web page in response to requests for said defined web page.

Claim 2 (original): The method of claim 1 wherein said web page is defined and stored in a centralized repository.

Claim 3 (previously presented): The method of claim 2 wherein defining said web page comprises the steps of:

- defining a placement and derivation for all elements comprising said web page;

and

- defining said web page as either static dynamic in nature.

Claim 4 (original): The method of claim 3 wherein said elements are defined as dynamic or static in nature.

Claim 5 (original): The method of claim 4 wherein the creation of said executable code and the creation of the scheduler component is generated from Business Class definitions comprised of said defined web pages.

Claim 6 (original): The method of claim 5 wherein said static copy of said defined web page is stored in a format capable of being viewed by a web browser.

Claim 7 (previously presented): A method for providing a requestor with access to dynamic data via quasi-static data requests, comprising the steps of:  
providing a web page including at least one dynamic element;  
a scheduler periodically invoking an executable to generate a quasi-static copy of the web page; and  
upon receiving a request for the web page from the requestor, returning the quasi-static copy of the web page.

Claim 8 (previously presented): The method of claim 7 wherein the at least one dynamic element is retrieved from an operational database by the executable when the quasi-static copy of the web page is generated.

Claim 9 (previously presented): The method of claim 7 wherein the executable is written in Active Server Pages (ASP).

Claim 10 (previously presented): The method of claim 7 wherein the quasi-static copy of the web page is stored in a format capable of being viewed by a web browser.

Claim 11 (previously presented): A system for providing a requestor with access to dynamic data via quasi-static data requests, comprising the steps of:  
a web page including at least one dynamic element;  
an executable that generates static copies of the web page; and  
a scheduler that periodically invokes the executable to generate a static copy of the web page, wherein the static copy of the web page is returned to the requestor as a quasi-static copy of the web page.

Claim 12 (previously presented): The system of claim 11 wherein the at least one dynamic element is retrieved from an operational database by the executable when the static copy of the web page is generated.

Claim 13 (previously presented): The system of claim 11 wherein the executable is written in Active Server Pages (ASP).

Claim 14 (previously presented): The system of claim 11 wherein the static copy of the web page is stored in a format capable of being viewed by a web browser.